

Software Requirement Specification

GROUP DISCUSSION PORTAL

NAME	Aanest Raja C
ROLL NO	7376222AD104
PROJECT ID	40
PROBLEM STATEMENT	Group Discussion Portal

Technology:LAMP STACK

Frontend	HTML, CSS, JS
Backend	LINUX, PHP with Laravel framework
Data Base	MYSQL

PROBLEM STATEMENT:

Build a portal for Group Discussion that allows a user to book a slot for attending GD which includes date and time, slot number, and venue.

Software Requirement Specification

PURPOSE

- The Group Discussion booking portal serves as a centralized platform for users to reserve slots for participating in Group Discussions (GD). The purpose of this project is to streamline the process of scheduling and managing GD sessions.
- Through the portal, users can select their preferred date and time, choose from available slot numbers, and pick a venue for the discussion.
- This system offers convenience and efficiency by eliminating the need for manual coordination and communication. It enhances user experience by providing a user-friendly interface that simplifies the booking process.

SCOPE OF PROJECT

- The Group Discussion booking portal project encompasses the development of a comprehensive platform enabling users to efficiently book slots for participating in group discussions.
- It will feature a user-friendly interface displaying available slots for GD sessions, allowing users to select their preferred date, time, and venue from the provided options. Upon booking, users will receive confirmation notifications, enhancing user experience and ensuring seamless interaction.
- Administrators will have access to a backend system for managing booked slots, monitoring attendance, and making necessary adjustments. This includes features for adding or removing slots, blocking out unavailable times, and generating reports for analysis.

Software Requirement Specification

SYSTEM OVERVIEW

User

- Users can register and login.
- View available slots for group discussions.
- Book a slot for a specific date, time, and venue.
- If a user booked the slot and not attended, it will be marked as Absent.

Faculty

- Edit or cancel booked slots.
- They can give Remarks.

Admin

- Admin panel can access everything in the DB.
- The Admin is responsible for confirming the Booking of venues .
- They can change slots,date,time,venue and can cancel the booked slot etc..
- Integration with email notifications for slot booking confirmation, reminders, and cancellations.

FEATURES

User Authentication:

- Users should be able to register with their email and password.
- Users should be able to login securely.

Slot Booking:

Software Requirement Specification

- Display available slots with date, time, Faculty, venue, and slot number.
- Allow users to select a slot and book it.

Faculty:

- They can only edit Time,Venue,Event type.
- They can give remarks and publish results.

Admin Panel:

- Admin should be able to add, edit and access everything in the DB
- Admin should manage all details.

Notifications:

- Email notifications for slot booking confirmation, reminders, and cancellations.

Registration Status:

- Students can view the current status of their Slot Registration in the registered event page.

Admin Access:

- Admin can view all the registered slots , view slot details, approve or reject the slots with suitable remarks, schedule Information status.

System Requirements Specification

Software Requirement Specification

Functional Requirements:

- **User Management:**

1. Students can register and login.
2. Admins have access control with an analytical dashboard and dedicated features .

- **Slot Booking:**

1. Users can book their slots with appropriate details and there will be a list of choices.
2. Slot Booking page contains,
 - Faculty
 - Event
 - Venue
 - Date and Time

- **Registration Status:**

1. Students can view the current status of their slot booking.
2. If registration is successful it will be shown on the selected event page.
3. If the registration is canceled then the remarks are shown.
4. Students can also see the logs of their registration.

- **Slot Scheduling:**

1. If the Registration is confirmed then slots will be allocated for the particular Event.

Non-Functional Requirements:

Software Requirement Specification

- **Performance:** The system must respond to user actions within 2 seconds to ensure efficient usability and must handle a concurrent user load of at least 500 users without significant performance degradation.
- **Security:** User data must be encrypted during transmission and storage, and access to sensitive functionalities should be restricted to authorized admin users through secure authentication mechanisms.
- **Usability:** The user interface should be intuitive and user-friendly, with clear and concise error messages provided to guide users in case of input errors or system failures.
- **Reliability:** The system should be available 24/7 with minimal downtime and should have a backup and recovery mechanism in place to prevent data loss in case of system failures or crashes.
- **Scalability:** The system should be designed to accommodate an increasing number of users and data volume over time, and it should be scalable to support additional features and functionalities as per future requirements.

DATABASE DESIGN

1. To create a database,

```
CREATE TABLE Users (  
    user_id INT AUTO_INCREMENT PRIMARY KEY,  
    username VARCHAR(255) NOT NULL,  
    email VARCHAR(255) NOT NULL UNIQUE,  
    password VARCHAR(255) NOT NULL  
);
```

Software Requirement Specification

```
CREATE TABLE Venues (  
    venue_id INT AUTO_INCREMENT PRIMARY KEY,  
    venue_name VARCHAR(255) NOT NULL,  
    capacity INT NOT NULL  
);  
  
CREATE TABLE Slots (  
    slot_id INT AUTO_INCREMENT PRIMARY KEY,  
    date DATE NOT NULL,  
    time TIME NOT NULL,  
    venue_id INT NOT NULL,  
    is_booked BOOLEAN NOT NULL DEFAULT FALSE,  
    FOREIGN KEY (venue_id) REFERENCES Venues(venue_id)  
);
```

- **Retrieve Available Slots for a Specific Date and Venue:**

```
SELECT slot_id, date, time  
FROM Slots  
WHERE date = '2024-04-30'  
AND venue_id = 1  
AND is_booked = 0;
```

- **Book a Slot for a User:**

```
UPDATE Slots  
SET is_booked = TRUE  
WHERE slot_id = 123;
```

- **Retrieve User's Booked Slots:**

```
SELECT Slots.slot_id, date, time, venue_name, venue_address  
FROM Slots
```

Software Requirement Specification

```
JOIN Venues ON Slots.venue_id = Venues.venue_id  
WHERE user_id = 456  
AND is_booked = TRUE;
```

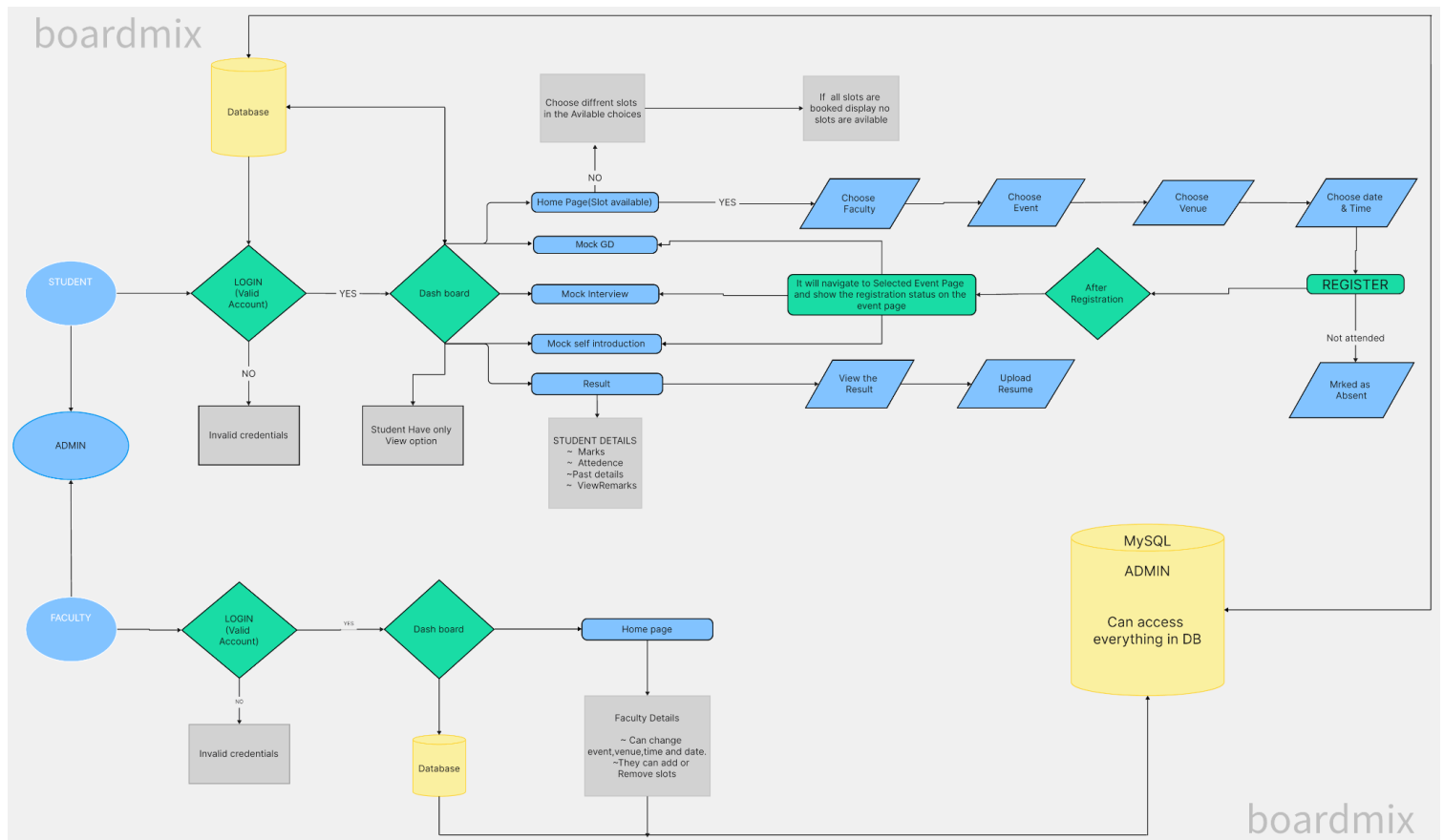
- **Add a New Venue:**

```
INSERT INTO Venues (venue_name, capacity)  
VALUES ('Conference Room A', 50);
```

- **Delete a Slot:**

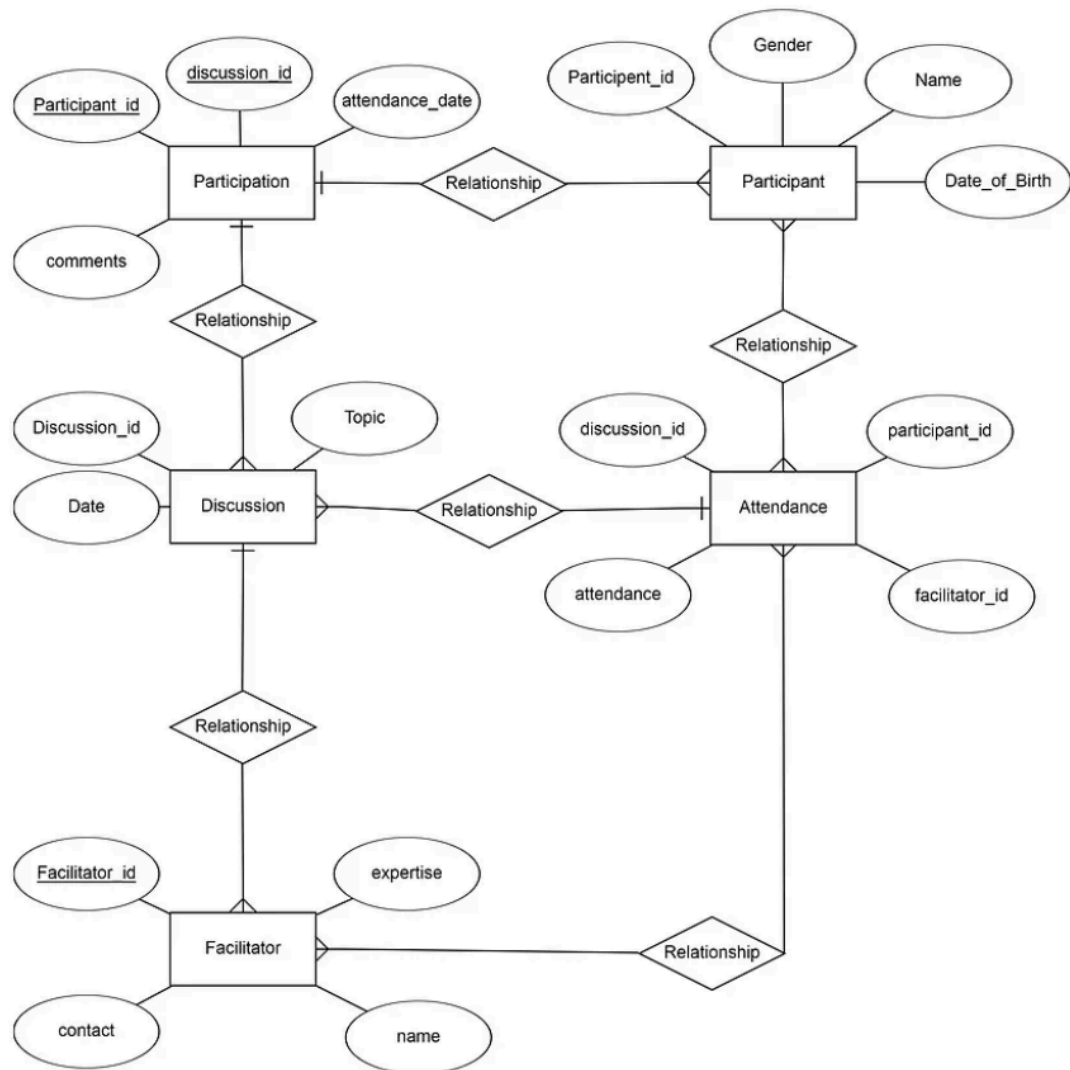
```
DELETE FROM Slots  
WHERE slot_id = 789;
```

FLOW CHART




Software Requirement Specification

ER DIAGRAM




Software Requirement Specification

PROJECT OUTCOME




Login

Username

 Enter Username



Password

 Enter Password

[Forgot Password?](#)

Login

Or Sign Up Using



[Home](#) [Mock GD](#) [Mock Interview](#) [Mock Self Introduction](#) [Result](#)

Welcome to Communication Skill Assessment

Practice Makes Perfect

Choose a Faculty

Faculty-2

Choose an Event

Mock GD

Choose a Venue

CSE-Conference Hall-1

Choose Date & Time

10/05/2024 & 8:30AM to 9:30AM

Submit

Software Requirement Specification

[Home](#) [Mock GD](#) [Mock Interview](#) [Mock Self Introduction](#) [Result](#)

About Mock GD

It is a structured conversation involving a group of individuals, usually between 6 to 15 participants, who gather to discuss a specific topic or problem.

Registration Successful

Date & Time:	10/05/2024 & 8:30AM to 9:30AM
Faculty:	Faculty-2
Event:	Mock GD
Venue:	CSE-Conference Hall-1

[Home](#) [Mock GD](#) [Mock Interview](#) [Mock Self Introduction](#) [Result](#)

Dashboard

Upload your resume:

No file selected.

Remarks by Faculty:

Results: